
Subject: Re: Sort a HASH

Posted by [chris_torrence@NOSPAM](#) on Fri, 24 Jun 2011 22:28:04 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi all,

The reason this isn't working is because when you index a hash, you get back another hash. It doesn't matter the order of the keys when you do the indexing, IDL will create a new hash with the desired key/value pairs, and they will be in whatever order the hash determines.

A way around this is to just get both the keys & values out, then sort on those:

```
h=hash(['a','c','d','b'],[7,1,5,8])
keys = (h.KEYS()).toArray()
val = (h.VALUES()).toArray()
s = SORT(keys)
print, keys[s], val[s]
```

```
a b c d
   7   8   1   5
```

If you were worried about making an extra copy of the values, you could use a foreach loop instead:

```
foreach key, (h.KEYS())[s] do print, key, h[key]
a      7
b      8
c      1
d      5
```

This will work in IDL 8.1.

Perhaps a better approach would be for us to add a SORT keyword to the KEYS() method. So you could do something like:

```
foreach key, h.KEYS(/SORT) do print, key, h[key]
```

What do you all think about that?

-Chris
ITTVIS
